

Politechnika Warszawska

W Y D Z I A Ł E L E K T R Y C Z N Y



Instytut Elektrotechniki Teoretycznej i Systemów Informacyjno-Pomiarowych

Praca dyplomowa magisterska

na kierunku Informatyka Stosowana
w specjalności Inżynieria Oprogramowania

Porównanie efektywności wybranych narzędzi służących do serwowania danych

inż. Jan Łukomski

numer albumu 291089

promotor

prof. dr hab. inż. Remigiusz Rak

WARSZAWA 2024

Porównanie efektywności wybranych narzędzi służących do serwowania danych

Streszczenie

TODO na całą stronę

Słowa kluczowe: A, B, C

Comparison of the effectiveness of selected data serving tools
Abstract

TODO

Keywords: X, Y, Z

Spis treści

1	Wstęp	9
1.1	Cel pracy i zakres pracy	9
2	Przegląd istniejących badań dla wybranych narzędzi	11
3	Opis koncepcji badania	13
3.1	Koncepcja wirtualnego użytkownika	13
3.2	Badanie pojedynczego zapytania	13
3.3	Badania limitu użytkowników	13
4	Opis wykorzystywanych narzędzi i bibliotek	15
4.1	Django	15
4.2	Dotnet	15
4.3	NestJS	15
4.4	K6	15
4.5	Docker	15
5	Przygotowanie aplikacji	17
5.1	Zbiory danych	17
6	Badanie aplikacji	19
6.1	Pojedyncze zapytania	19
6.2	Limity równoległych zapytań	20
7	Podsumowanie i wnioski	21
	Bibliografia	23
	Wykaz skrótów i symboli	23
	Spis rysunków	25
	Spis tablic	27

Rozdział 1

Wstęp

1.1 Cel pracy i zakres pracy

Rozdział 2

Przegląd istniejących badań dla wybranych narzędzi

Rozdział 3

Opis koncepcji badania

3.1 Koncepcja wirtualnego użytkownika

W kontekście testów obciążeniowych, wykorzystywanie wirtualnych użytkowników jest powszechną praktyką mającą na celu symulowanie działania rzeczywistych użytkowników na aplikacji. Koncepcja ta odnosi się do abstrakcji wirtualnych jednostek, które emulują interakcje z aplikacją podczas testów. To podejście pozwala na ocenę wydajności systemu w warunkach obciążeniowych, gdzie liczba użytkowników korzystających z aplikacji może znacząco wzrosnąć.

Dzięki możliwości kontrolowania liczby wirtualnych użytkowników, testy obciążeniowe umożliwiają programistom identyfikację potencjalnych problemów związanych z wydajnością, takich jak opóźnienia w odpowiedziach serwera czy niestabilność systemu podczas dużego natężenia ruchu. Symulowanie różnych scenariuszy użycia za pomocą wirtualnych użytkowników pozwala na zrozumienie, jak aplikacja radzi sobie z różnymi obciążeniami, co z kolei umożliwia optymalizację i dostosowanie systemu do rzeczywistych warunków użytkowania. W praktyce, dzięki temu można przetestować skrajną sytuację nagłego wzmożenia ruchu na stronie i być gotowym na taki scenariusz.

3.2 Badanie pojedynczego zapytania

3.3 Badania limitu użytkowników

Rozdział 4

Opis wykorzystywanych narzędzi i bibliotek

4.1 Django

4.2 Dotnet

4.3 NestJS

4.4 K6

4.5 Docker

Rozdział 5

Przygotowanie aplikacji

5.1 Zbiory danych

W celu przeprowadzenia badania konieczne było przygotowanie odpowiednich zbiorów danych, które miały posłużyć do symulacji różnych scenariuszy. W tym kontekście przygotowano dwa zbiory danych, aby umożliwić różnorodne analizy:

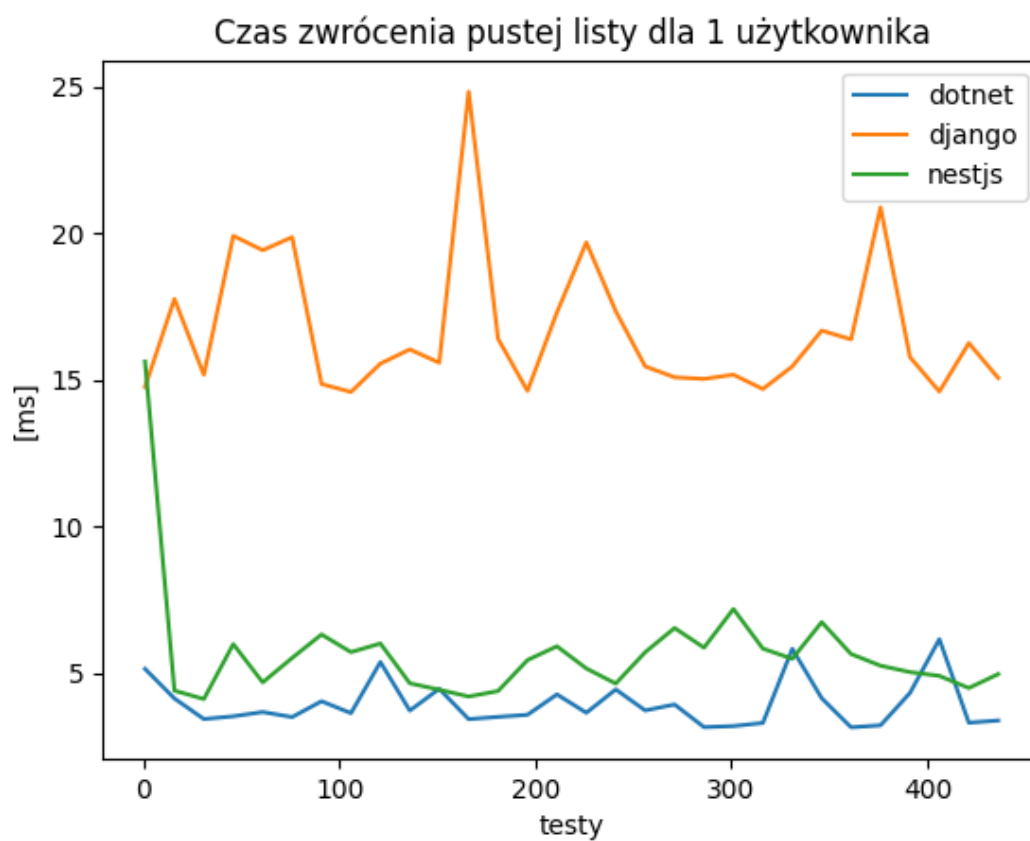
- **FWB_0** - Jest to zbiór pusty, pozbawiony jakichkolwiek elementów. Brak danych w tym zbiorze ma posłużyć do sprawdzenia zachowania systemu w sytuacji, gdy nie ma żadnych rekordów do przetworzenia.
- **FWB_100K** - Ten zbiór składa się z 100 000 elementów. Każdy element tego zbioru reprezentuje pojedynczy rekord w bazie danych i zawiera unikalne identyfikatory (numery) oraz nazwy (tekstowe). Zbiór ten został przygotowany w celu przetestowania wydajności systemu oraz jego reakcji na duże ilości danych.

Jeden element to rekord w bazie danych zawierający ID (number) oraz nazwę (tekst). Przygotowanie tych zbiorów danych stanowiło niezbędny krok przed przystąpieniem do właściwej analizy i symulacji różnych scenariuszy w badaniu. Dzięki tym zbiorom możliwe było zbadanie zachowania systemu w różnych warunkach oraz przeprowadzenie odpowiednich wniosków na podstawie uzyskanych wyników.

Rozdział 6

Badanie aplikacji

6.1 Pojedyncze zapytania



Rysunek 1. Czas zwrócenia pustej listy dla 1 użytkownika

Zdecydowanym liderem zestawienia jest Dotnet

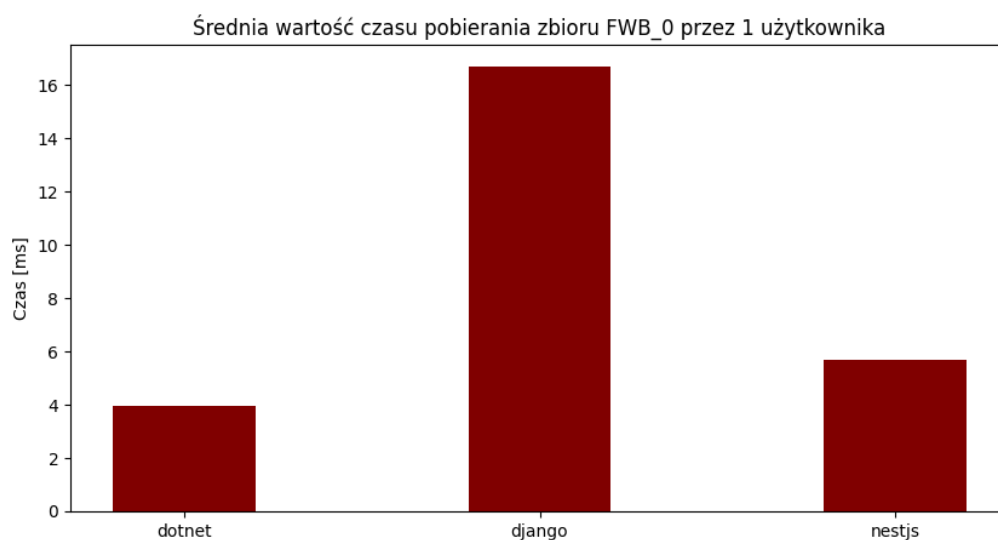
6.2 Limity równoległych zapytań

Narzędzie	Liczba użytkowników (FWB_0)	Liczba użytkowników (FWB_100K)
Django	3159	3070
Dotnet	14469	90
NestJS	4132	0

Django potrafi obsługiwać podobną ilość użytkowników dla obu zbiorów. W przypadku zbioru danych FWB_100K, narzędzie zanotowało spadek o niecałe 3%.

Dotnet jest zdecydowanym liderem liczby użytkowników dla pustego zbioru. Widać u niego jednak znaczny spadek liczby obsługiwanych użytkowników wraz ze wzrostem ilości przesyłanych danych.

NestJS które przy zbiorze pustym jest nieco lepsze od Django, dla zbioru FWB_100K całkowicie przestało odpowiadać - co oznacza, że nie radzi sobie z taką ilością danych.



Rysunek 2. Średni czas zwrócenia pustej listy dla 1 użytkownika

Rozdział 7

Podsumowanie i wnioski

Wykaz skrótów i symboli

Spis rysunków

1	Czas zwrócenia pustej listy dla 1 użytkownika	19
2	Średni czas zwrócenia pustej listy dla 1 użytkownika	20

Spis tablic

Spis załączników

1	Dowód próżni doskonałej.....	31
2	Dowód zera bezwzględnego	33
3	Dowód czasu zatrzymanego	35
4	Dowód nieskończoności urojonej	37

Załącznik 1

Dowód próżni doskonałej

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Załącznik 2

Dowód zera bezwzględnego

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Załącznik 3

Dowód czasu zatrzymanego

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

Załącznik 4

Dowód nieskończoności urojonej

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.